**CMSC214 Project 2**

**Content: Project 2 |**

**Lessons Learned, Screenshots, Checklist**

**Class: CMSC214 - CRN: 22626**

**Instructor: Mark Estep**

**Description: A program in JavaFX that creates a car shape and simulates a car race in an animation.**

**Due: 09/20/2020**

**Student name: Derya Ozdemir Kurin**

**MC ID: M21091005**

**Lessons Learned:**

During this project, I had the chance to apply what I learned in chapter 14 and 15.

I first figured it out how to create a car shape with using rectangle, polygon and two circles. I created a CarPane class that extended from Pane and defined all the necessary data fields there.

In the CarPane class, I included a formCar method that created the shapes and after coloring them added to the created pane. At the beginning of the formCar method, I had to clear all the child nodes (shapes that created the car) of the object each time it was called, otherwise all the cars created during the animation were visible and it colored the bottom part of the pane along the line, the car shape was not clear.

In this class I defined four handler methods for pausing, playing, increasing and decreasing the animation rate.

This class had a constructor method which first created the car shape, then called a called moveCar handler in a KeyFrame object in Timeline class to create an animation.

In moveCar I set the coordinates of the CarPane object for continues iteration and when it reached the width of the pane it is location, I set the x coordinates to 0 again in order to have a continues movement of the car.

I also created buttons and registered the handlers I defined in CarPane class. Later I wanted to show the car in the bottom of the pane and buttons on the top.

To achieve this, I utilized three panes in total;

HBox for buttons, (I first tried to use GridPane to place the buttons with some padding between them, but it did not work).

CarPane to keep all the car race animation in it.

BorderPane to place CarPane object and the Buttons in HBox.

I first could not see the car when running the program, but after some research I found the requestFocus() method to make the animation visible to the pane.

The buttons worked immediately but I realized pressing the up and down keys did not change anything. I then figured it out that I need to register them on the BorderPane object not the CarPane.

In addition to the project, I created mouse pressed and mouse released events for extra practice. With them too, I needed to register these events to the most outer pane BorderPane. When I experimented the mouse event with my MacBook’s touchpad, I could stop the race animation by clicking the touchpad but releasing the click did not seem to work for me. However, I believe that with an ordinary mousepad, it should work since the implementation was straight forward.

I think this exercise was great to practice everything in the two chapters.

I feel more confident with created grouped shapes and using them in an animation and manipulate the rate of the animation with various event handlers. In addition I made use of online sources and Oracle JavaFx documentation.

**Screenshots:**

A screenshot of a cell phone

Description automatically generated

A screenshot of a social media post

Description automatically generated

**Checklist:**

|  |  |  |  |
| --- | --- | --- | --- |
| **#** |  | **Y/N** | **Comments** |
|  | Source java files | **Y** |  |
|  | Files: |  |  |
|  | LastNameFirstinitial\_Project02.zip | **Y** |  |
|  | LastNameFirstInitial\_Project02.doc | **Y** |  |
|  | Program compiles | **Y** |  |
|  | Program runs | **Y** |  |
|  | Checklist is completed and included in the Documentation |  |  |
|  | Documentation file: |  |  |
|  | ~~Comprehensive Test Plan~~ |  |  |
|  | Screenshots of running program | **Y** |  |
|  | ~~UML diagram~~ |  |  |
|  | Lessons Learned | **Y** |  |
|  | Checklist | **Y** |  |